Motionnet
RemoteI/O & RemoteMotion

Motionnet

DLLサンプル説明書

NPM
Nippon Pulse Motor Co., Ltd.

「ご使用に際して」

●弊社以外から購入の場合
弊社以外から製品を購入された場合の保証につきましては、購入先にお問い合わせください。

●保証期間
保証期間は、製品をご指定の場所に納入後、一年間と致します。

●保証範囲
本書に従った正常な使用状態のもとで、保証期間内に故障が発生した場合は、無償で修理を致します。
但し、保証期間内であっても、故障の原因が次のいずれかに該当する場合は、無償修理の対象外とさせて頂きます。
①当社又は当社が指定した者以外による改造又は修理に起因する場合
②納品後の落下、運送上での損傷に起因する場合
③製品の部品の自然劣化、摩耗又は疲労等による場合
（モータ軸受、歯車、グリス、ケーブル類等）
④当社製品の本来の使い方以外の使い方をしたことに起因する場合
⑤火災、地震、落雷、風水害、塩害、電圧異常その他の天災又は不可抗力に起因する場合
⑥その他、故障の原因が、当社の責とみなされない事由に起因する場合
注１）日本国外に輸出された製品に関しては、保証の対象外とさせて頂きます。
注２）無償修理は、弊社持ち込み修理のみとし、出張修理等は致しません。
注３）修理が行われた製品の保証期間は、故障前の保証期間を超えることなく、修理前の保証期間と同一になります。
注４）ここでの保証は、製品単体の保証を意味するものであり、製品の故障等により誘発される損害は保証の対象外とさせて頂きます。
注５）弊社の判断により、修理に代えて、交換とさせて頂く場合がございます。

●本書は、製品に含まれる機能詳細を説明するものであり、お客様の特定目的に適合することを保証するものではありません。
また、本取扱説明書に記載されている応用例、回路図等は参考用です。ご採用に際しては、機器・装置の機能や安全性をご確認のうえ、ご使用ください。

●弊社製品は、原則として、次のいずれかの用途にはご使用なさらないようお願い致します。ご使用を要望される場合には、必ず、弊社営業部にご相談ください。
①原子力設備、電力・ガス等の供給システム、交通機関、車両設備、各種安全装置、医療機器等の高い信頼性・安全性を必要とされる設備
②人命や財産に直接、危険が及ぶ可能性がある設備
③カタログ、取扱説明書等に記載のない条件や環境での使用

●弊社製品の故障により人命や財産に重大な損害が及ぼす可能性のある用途にご使用する場合には、冗長設計等により、必ず高い信頼性・安全性を確保してご使用ください。

目　　次

１．はじめに.....1
２．仕様.....1
　２－１．対応するメインボードと DLL ファイル.....1
　２－２．動作 OS 1
　２－３．開発環境.....1
　２－４．注意.....1
３．構成.....2
４．準備.....3
5. サンプルプログラムの使用法と内容 4
　５－１．I/O.....4
　５－２．ERR.....5
　５－３．M1.....6
　５－４．M4.....7
　５－５．6045.....9

１．はじめに

このたびは、超高速シリアル通信システム Motionnet シリーズのセンターボードをお求めいただきまして、ありがとうございます。

本取扱説明書は、Motionnet DLL を使用したサンプルプログラムについて解説しております。ご利用にあたっては「DLL リファレンス」などを参照ください。

２．仕様

２－１．対応するメインボードと DLL ファイル

メインボード	DLL ファイル
Motionnet センターボード(PCI) PPCI-L112	PPCI_L112.DLL(32bit 用) PPCI_L112_X64.DLL(64bit 用)
Motionnet センターボード(PC/104) NPMCMNET-I/0104	NPMCMNET104.DLL(32bit 用)
Motionnet センターユニット(USB) MNET-PUSB3601	MNETPUSB_X86.DLL(32bit 用) MNETPUSB_X64.DLL(64bit 用)

２－２．動作 OS

Windows2000, WindowsXP, WindowsVista, Windows7, Windows8

２－３．開発環境

サンプルプロジェクトは以下の環境で作成しています。
その他の環境でのご利用にあたりましては、変換ウィザードなどを御利用下さい。

サンプル種類	開発環境	対応環境
VB	Visual Basic 6.0	32
VC	VisualC++ 6.0	32
VBNET	Visual Studio 2003	32/64
CS(C#)	Visual Studio 2005	32/64
VC2005	Visual Studio 2005	32/64

２－４．注意

（1）サンプルプログラムの記述はあくまで一例です。プログラム開発にあたっては DLL リファレンスなどを参照ください。
（2）モータなどを接続しての動作試験においては、予想外の動作となった場合でも危険のないよう、非常停止入力やハードエンドリミット入力を接続するなどし、それらの信号入力によるパルス出力停止を確認した上で行ってください。
（3）関数の使用手順/使用方法，動作の実現方法を説明するというサンプルプログラム本来の目的から不正入力などの各種イレギュラー処理は省いてありますのでご注意ください。
（4）デバイスドライバ、DLL、ローレベル DLL のファイルは同じパッケージ(CD-R など)で提供された組合せで御利用下さい。

３．構成

サンプルプログラムは各センターボード毎に以下のように構成されております。

[Sample]
- [IO]　最も簡単な通信開始から I/O の入出力まで
  - [VC]
  - [VB]
  - [VBNET]
  - [CS]
  - [VC2005]
- [ERR]　上記[IO]のプログラムに通信エラーのモニタとクリアを付加
  - [VC]
  - [VB]
  - [VBNET]
  - [CS]
  - [VC2005]
- [M1]　1 軸モーションボードの制御
  - [VC]
  - [VB]
  - [VBNET]
  - [CS]
  - [VC2005]
- [M4]　4 軸(6143)モーションボードの制御
  - [VC]
  - [VB]
  - [VBNET]
  - [CS]
  - [VC2005]
- [6045]　4 軸(6045)モーションボードの制御
  - [VC]
  - [VB]
  - [VBNET]
  - [CS]
  - [VC2005]

注意

4 軸モーションボード基板型式区分

[４軸(6143)モーションボード]

・ MNET-BCDC5030A4

・ MNET-BCD4020FUA4

[４軸(6045)モーションボード]

・ MNET-M204-DUM

４．準備

（１） センターボードのデバイスドライバをインストールします。
（２） 目的のサンプルプログラムファイルを任意のフォルダにコピーします。
（３） コピーしたファイルの読み取り専用属性を解除してください
（エクスプローラでファイルを選択してプロパティ、チェックマークをはずして適用）
（４）開発環境別の準備

[Visual C++(以下 VC)の場合]
◆[VC]フォルダに Lib ファイルとヘッダーファイルをコピーしてください。
◆実行ファイルが生成されるフォルダ(下層の[Debug]フォルダなど)に DLL ファイルを
コピーしてください。
(USB の場合はローレベル DLL ファイル(MNETUSBL_X86.DLL/ MNETUSBL_X64.DLL)もコピーください)

[Visual Basic の場合]
◆[VB]フォルダに bas ファイルと DLL ファイルをコピーしてください。
(USB の場合はローレベル DLL ファイル(MNETUSBL_X86.DLL)もコピーください)

[VB.NET の場合]
◆[VBNET]フォルダに vb ファイルをコピーしてください。
(関数定義用 vb ファイルは 32bit 用/64bit 用共ファイル名は同じですが、一部内容が異なりますのでご注意ください。(各関数が参照する DLL ファイル名記述)
◆実行ファイルが生成されるフォルダ(下層の[bin]フォルダなど)に DLL ファイルを
コピーしてください。
(USB の場合はローレベル DLL ファイル(MNETUSBL_X86.DLL/ MNETUSBL_X64.DLL)もコピーください)

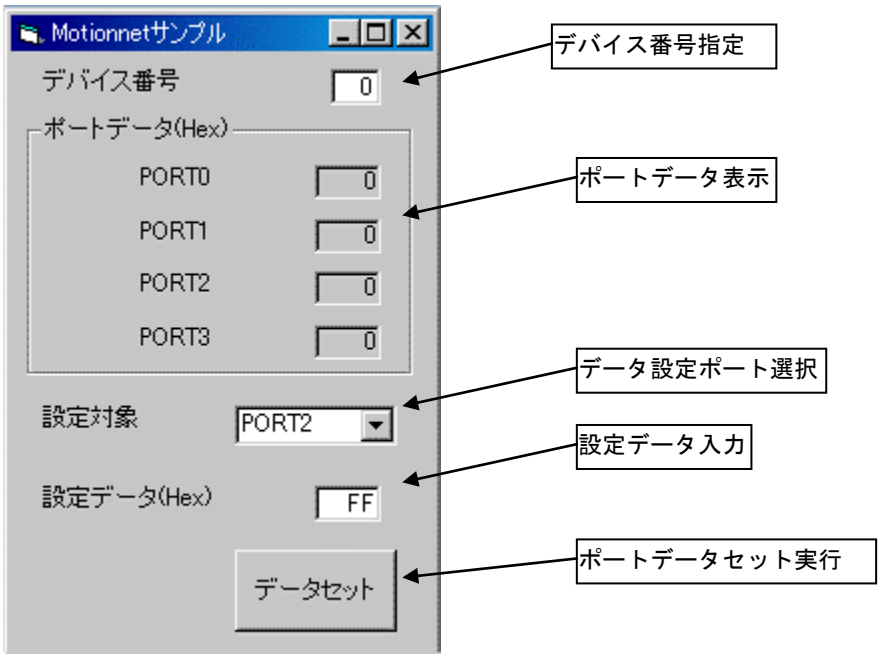
[C#の場合]
◆[CS]フォルダに cs ファイルをコピーしてください。
(関数定義用 cs ファイルは 32bit 用/64bit 用共ファイル名は同じですが、一部内容が異なりますのでご注意ください。(各関数が参照する DLL ファイル名記述)
◆実行ファイルが生成されるフォルダ(下層の[bin]フォルダなど)に DLL ファイルを
コピーしてください。
(USB の場合はローレベル DLL ファイル(MNETUSBL_X86.DLL/ MNETUSBL_X64.DLL)もコピーください)

[VC2005 の場合]
◆[VC2005]フォルダに Lib ファイルとヘッダーファイルをコピーしてください。
◆実行ファイルが生成されるフォルダ(対象環境下層の[Debug]フォルダなど)に DLL ファイルを
コピーしてください。
(USB の場合はローレベル DLL ファイル(MNETUSBL_X86.DLL/ MNETUSBL_X64.DLL)もコピーください)
◆同じプロジェクト内に 32bit 用(Win32),64bit 用(x64)両方の設定が保存してありますので、
対象のプラットフォームを選択してご利用ください。(下図参照)

5. サンプルプログラムの使用法と内容

５－１. I/O

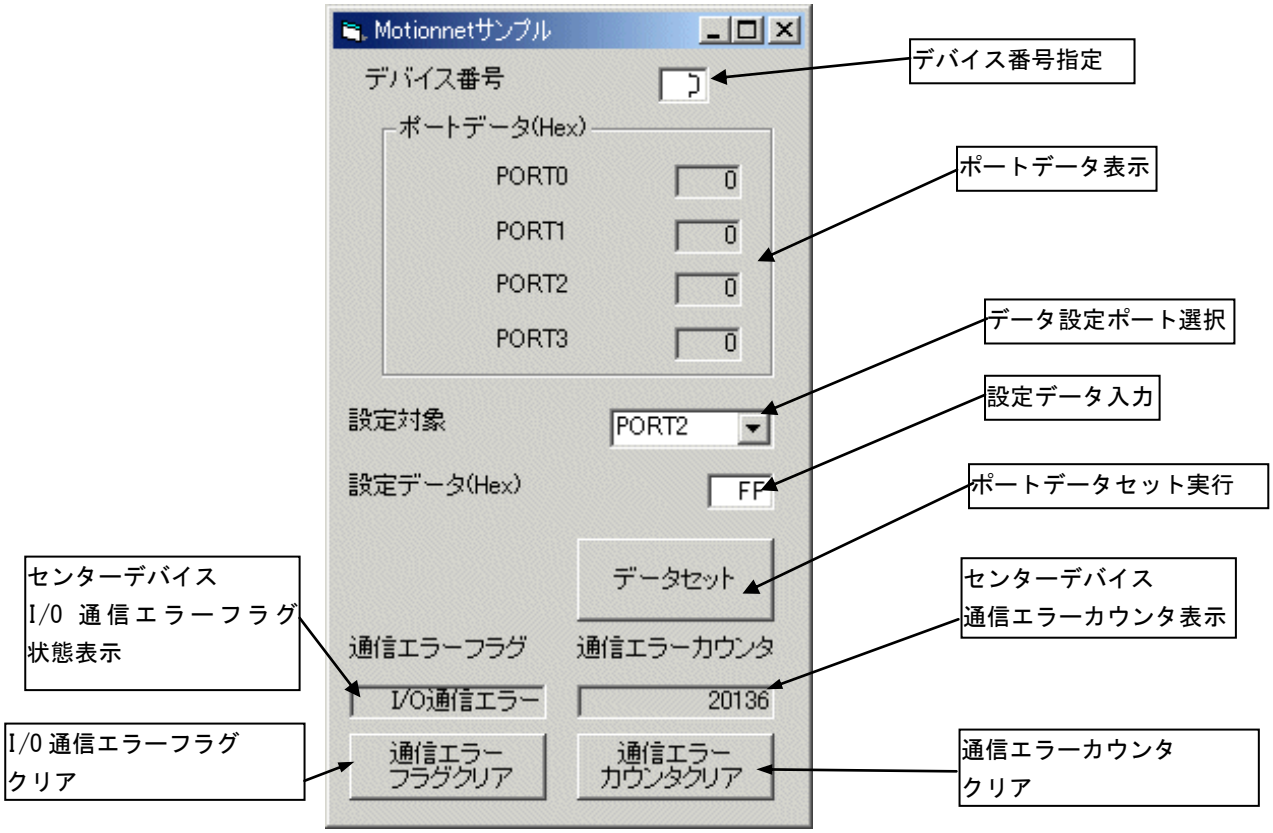
最も簡単な通信開始から I/O の入出力処理が記述されています。
サンプルプログラムを起動し、正常にボードをオープンなどの初期処理が完了すると VC, VB の場合共に下図のような操作ダイアログが表示されます。(画面は VB の例)



(1) 0 ライン目に I/O ボードを接続し電源を入れてください。
ライン番号はプログラム記述により 0 ライン目固定, デバイス番号はフォーム上部のテキストボックスで指定します。(デバイス番号は 0 始まりです)
(2) フォームロードで初期処理(ボードオープン、センターデバイスリセット、システム通信、I/O 通信開始)が行われ通信が開始されます。
(3) タイマーで、I/O ポートのデータが読み出され、表示されます。
(4) データセットのクリックで、選択したポートにデータが設定されます。

５－２．ERR

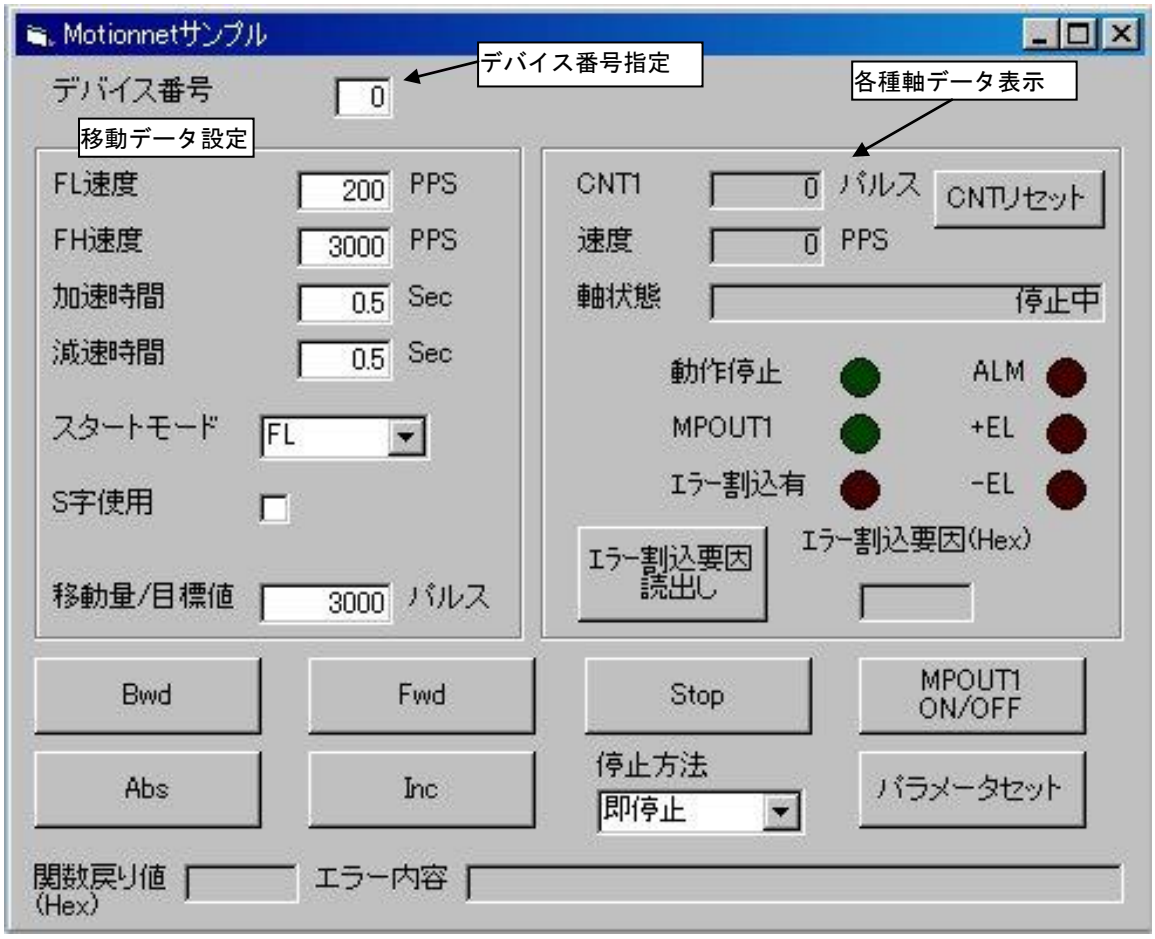
前述の最も簡単な I/O のプログラムに I/O 通信エラー(フラグ)、およびセンターデバイスの通信エラーカウンタのモニタとクリア処理を付加しています。
サンプルプログラムを起動し、正常にボードをオープンなどの初期処理が完了すると VC,VB の場合共に下図のような操作ダイアログが表示されます。(画面は VB の例)
これはセンターデバイスの I/O 通信エラーフラグが ON、センターデバイスの通信エラーカウンタがカウントアップしている場合の状態です。



(1) 0 ライン目に I/O ボードを接続し電源を入れてください。
ライン番号はプログラム記述により 0 ライン目固定, デバイス番号はフォーム上部のテキストボックスで指定します。
(2) フォームロードで初期処理(ボードオープン、センターデバイスリセット、システム通信、I/O 通信開始)が行われ通信が開始されます。
このとき、通信エラーカウンタに約 250 前後の((64-接続デバイス数)×4)通信エラーがカウントされますが、これはライン上に接続される全 64 デバイスに対しシステム通信を行った際、接続されていないデバイスへのシステム通信が無応答となるために起こる事象であり、通信品質などの問題ではありません。
(3) タイマーで、I/O ポートのデータ、センターデバイスの I/O 通信エラーフラグ、通信エラーカウンタ状態が読み出され、表示されます。
(4) データセット のクリックで、選択したポートにデータが設定されます。
(5) 通信エラーフラグクリア のクリックで、I/O 通信エラーフラグのクリア処理が行われます。
(6) 通信エラーカウンタクリア のクリックで、通信エラーカウンタのクリア処理が行われます。

５－３．M1

1 軸モーションボード用のサンプルです。
サンプルプログラムを起動し、正常にボードをオープンなどの初期処理が完了すると VC, VB の場合共に下図のような操作ダイアログが表示されます。(画面は VB の例)



(1)0 ライン目に 1 軸モーションボードを接続し電源を入れてください。
ライン番号はプログラム記述により 0 ライン目固定, デバイス番号はフォーム上部のテキストボックスで指定します。
(2)フォームロードで初期処理(ボードオープン、センターデバイスリセット、システム通信、I/O 通信開始)が行われ通信が開始されます。
(3)タイマーで、各種軸データが読み出され、表示されます。
(4)軸動作などを行う前にまずパラメータセットをクリックしてください、信号論理や信号タイプの選択などのパラメータが設定されます。
パラメータ設定データはパラメータセットボタンのクリックイベントのコードの記述を確認し、必要があれば変更ください。
(5) 本 DLL 関数では「速度データ設定」と、「動作起動」は別関数になっており、それらは別々に行うことができますが、このサンプルプログラムでは、各種軸動作指令のボタンで「速度データ設定」と「動作起動」を一緒に行っています。
(6)このサンプルプログラムに組み込まれたほとんどの DLL 関数の戻り値とその内容を示すテキストがフォーム下部に表示されます。
(7)一部コード簡略化のため戻り値の確認を省いている箇所がありますのでご注意ください。

５－４．M4

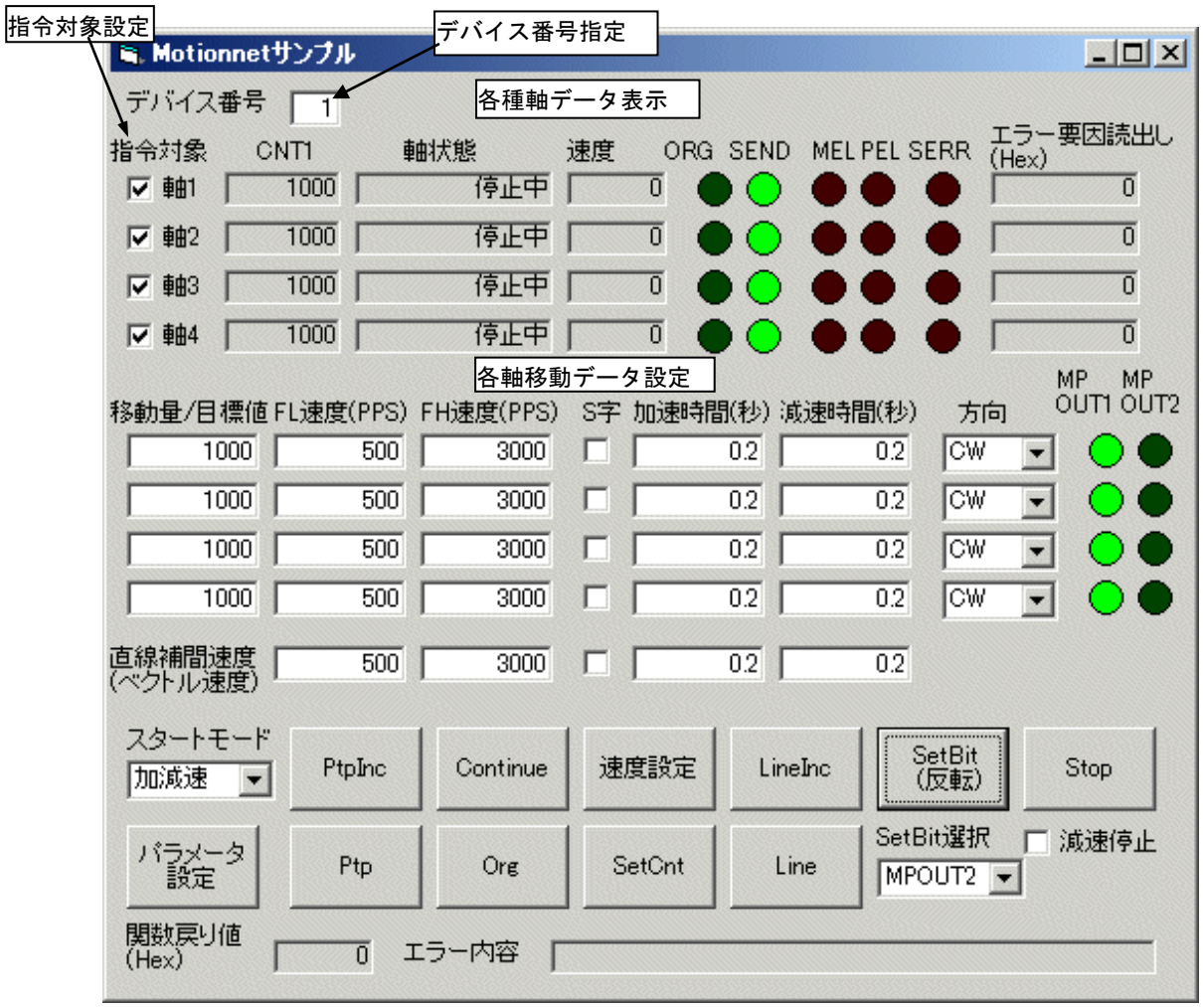
4 軸(6143)モーションボード用のサンプルです。
サンプルプログラムを起動し、正常にボードをオープンなどの初期処理が完了すると VC，VB の場合共に下図のような操作ダイアログが表示されます。(画面は VB の例)

(1)0 ライン目に 4 軸モーションボードを接続し電源を入れてください。
ライン番号はプログラム記述により 0 ライン目固定，デバイス番号はフォーム上部のテキストボックスで指定します。
(2)フォームロードで初期処理(ボードオープン、センターデバイスリセット、システム通信、I/O 通信開始)が行われ通信が開始されます。
(3)タイマーで、各種軸データが読み出され、表示されます。
(4)軸動作などを行う前にまずパラメータ設定をクリックしてください、信号論理や信号タイプの選択などのパラメータが設定されます。パラメータ設定データはパラメータ設定ボタンのクリックイベントのコードの記述を確認し、必要があれば変更ください。
(5) 指令の対象となる軸は画面左部のチェックボックスで指定します。
(6) PtpInc などの直線補間(Line/LineInc)動作以外の場合は、動作指令前に速度設定ボタンをクリックし、速度データを設定する必要があります。(直線補間(Line/LineInc)移動実行により前回設定された速度データは無効となります。また加減速動作を行なう場合はスタートモード: 加減速での速度設定が必要です)

(7) このサンプルプログラムに組み込まれたほとんどの DLL 関数の戻り値とその内容を示すテキストがフォーム下部に表示されます。
(8) 一部コード簡略化のため戻り値の確認を省いている箇所がありますのでご注意ください。

５－５．6045

4 軸(6045)モーションボード用のサンプルです。
サンプルプログラムを起動し、正常にボードをオープンなどの初期処理が完了すると VC，VB の場合共に下図のような操作ダイアログが表示されます。(画面は VB の例)



(1)0 ライン目に 4 軸モーションボードを接続し電源を入れてください。
ライン番号はプログラム記述により 0 ライン目固定，デバイス番号はフォーム上部のテキストボックスで指定します。
(2)フォームロードで初期処理(ボードオープン、センターデバイスリセット、システム通信、I/O 通信開始)が行われ通信が開始されます。
(3)タイマーで、各種軸データが読み出され、表示されます。
(4)軸動作などを行う前にまずパラメータ設定をクリックしてください、信号論理や信号タイプの選択などのパラメータが設定されます。パラメータ設定データはパラメータ設定ボタンのクリックイベントのコードの記述を確認し、必要があれば変更ください。
(5) 指令の対象となる軸は画面左部のチェックボックスで指定します。
(6) PtpInc などの直線補間(Line/LineInc)動作以外の場合は、動作指令前に速度設定ボタンをクリックし、速度データを設定する必要があります。(直線補間(Line/LineInc)移動実行により前回設定された速度データは無効となります。また加減速動作を行なう場合はスタートモード: 加減速での速度設定が必要です)

(7) このサンプルプログラムに組み込まれたほとんどの DLL 関数の戻り値とその内容を示すテキストがフォーム下部に表示されます。
(8) 一部コード簡略化のため戻り値の確認を省いている箇所がありますのでご注意ください。

注意	性能、品質の向上等にともない、お断り無しに掲載事項を変更させていただくことがありますので、あらかじめご了承ください。

NPM 日本パルスモーター株式会社

本　　社　〒113-0033　東京都文京区本郷 2-16-13　電話　03-3813-8841(代)　FAX.03-3813-8550
大阪営業所　〒552-0007　大阪市港区弁天 1-2-1-2314　電話　06-6576-8330　FAX.06-6576-8335
ＵＲＬ：　http://www.pulsemotor.com/

２０１４年４月発行